SoftMotion: DriveInterface: LenzeCAN

Last update: 16.04.2007

Hardware interface	CAN; must support 3S_CANdrv.lib
Supported drives	LenzeECSxM, Lenze 9400 Stateline
Runtimes	any
Author	Edwin Schwellinger/Hilmar Panzer
Components	LenzeCANDrive.lib; 3S_CanDrv.lib; SM_CAN.lib; SysLibCallback.lib; SysLibFile.lib
Version	1.9.3.0

CONTENT

1 Parameters in PLC config

1.1 BusInterface

wParam1	Not used
wParam2	Not used
dwParam1	Not used
dwParam2	Not used

1.2 AxisGroup

wParam1	CAN channel No (typically 0)
wParam2	Baudrate in kBit (125, 250, 500, 1000)
wParam3	SYNC generator: 0: PLC generates SYNC (only possible if PLC is highly precise); 1: first drive of AxisGroup generates SYNC 2: SYNC device generates SYNC (additional hardware needed)
wParam4	Not used
dwParam1	Reserved
dwParam2	Reserved
dwParam3	Not used
dwParam4	Not used

1.3 supported Drive.wControlType

1.3.1 ECS

The cyclic send data must consist of: fSetPosition.

The cyclic receive data can consist of: fActPosition.

1.3.2 9400 Stateline

The cyclic send data must consist of: fSetPosition, fSetVelocity, fSetTorque

The cyclic receive data can consist of: fActPosition, fActVelocity, fActTorque, fActCurrent.

1.4 Additional structure *LenzeECS_AXIS_REF*

name	Type	
byDriveState	BYTE	internal use: state of drive
wStatusWord	WORD	Status word C150 (ECSxM) / 16#6041 (9400)
wStatusWord2	WORD	Status word2 C155 (only ECSxM)

wControlWord	WORD	Control word C135 (ECSxM) / 16#6040 (9400)
wStatusWordOld	WORD	internal use
dwSetPosition	DWORD	set position sent to drive (RO)
byStatusInfo	BYTE	Bit8-11 of status word (ECSxM)
byErrAcknCounter	BYTE	Internal use
bHWLimitsActive	BOOL	TRUE, if one limit switch is active (ECSxM)
fLimitSwitchDeceleration	LREAL	Deceleration for stopping on limit switch (if 0, no ramp is applied) (ECSxM)
mcstopLS	MC_Stop	Internal use
strConfigFile	STRING	full name and path of config file
acit	…	internal use
txList, rxList, txPDOs, rxPDOs	…	internal use
byActOpMode	BYTE	Object 16#6061 (9400)
bySetOpMode	BYTE	Object 16#6060 (RO) (9400)
bOldReference	BOOL	Internal use
byDigitalInputs, byDigitalOutputs	BYTE	Digital inputs (16#60FD) and outputs (16#60FE/1) (only 9400)
crap	SMC_CANReadAllParams	internal use
pParameterlist	POINTER TO CAN_InitTelegram	internal use
wp	LenzeECSWriteParameter	internal use
rp	LenzeECSReadParameter	internal use

2 Features

- **RegulatorOn, DriveStart**
- Detecting and acknowledging **errors**
- **reading/writing** SoftMotion and **drive parameters** (to access index 0xaabb subindex 0xcc with length 0xdd in byte (only necessary for writing) either use MC_Read/Write(Bool)Parameter with parameter number -16#ddaabbcc) or better use specific FBs LenzeECSReadParameter and LenzeECSWriteParameter to access the Lenze Code positions directly or SMC_ReadCANParameter and SMC_WriteCANParameter to address a standard CAN object via index, subindex.
- **reading drive string parameters** with LenzeECSReadString
- any **gearing factors** (dwRatioTechUnitsDenom/iRatioTechUnitsNum)
- **linear/rotary axes**
- **controlling modes**: position, velocity (9400), torque (9400). Use SMC_SetControllerMode for switching.
- drive internal **homing** (first configure C3010, C0935, C0936 for ECSxM or 16#6098, 16#6099, 16#609A for 9400)
  Note (only ECSxM): during homing, the actual position is not reported from the drive!
- **latching**: 1 channel (TriggerNumber = 1) (only ECSxM)
- ECSxM: depending on C3175 **hardware limit switches** are handled by the controller (=3) or by the drive (else)
- **configuration from file**
- **configuration from dialogs in PLC config**
- supported **SYNC generators** (to be set in PLC Configuration, AxisGroup) : PLC, 1st drive, SYNC-Device

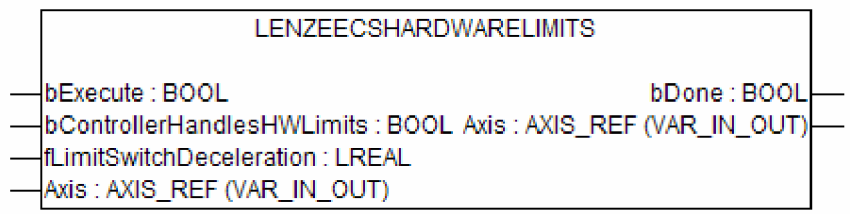
2.1 hardware limits (ECSxM)

The reaction on hardware limits is configured with C3175 inside the drive. If C3175 <> 3 the drive reacts on hardware limits. Then, the software might be unable to move an axis out of its limits, as the drive blocks this operation. Only if C3175 = 3, the drive is not handling the hardware limits, and only the controller handles them:

If bControllerHandlesHWLimits is set (default) and the axis runs on a hardware limit, the controller stops the axis and transfers it to errorstop. After a MC_Reset is done on the drive, it can be moved in the direction heading out of the limit switch; the other direction is blocked and produces an error as soon as the drive tries to move in this direction.

Depending on the deceleration value, the stop is done with a ramp (fLimitSwitchDeceleration>0) or immediately (else; default).

To set the values involved in this, a special FB is provided in LenzeECSdrive.lib:



With a rising edge on bExecute the new values are set.

3 configured parameters during startup

The following parameters are set during startup:

3.1 ECSxM

parameter	value set during startup		
	SYNC generator: PLC or SYNC-device	SYNC generator: 1st drive	
C352	0	0	CAN slave
C353/1	0	0	Address mode
C356/3	cycle time	cycle time	PDO cycle
C356/4	0	0	Delay time PDO
C1120	1	0	SYNC mode
C1121	cycle time	cycle time	SYNC cycle
C367	128	129	SYNC Rx ID
C368	128	128	SYNC Tx ID
C1123	cycle time/2	cycle time/2	SYNC window
C369	0	cycle time	SYNC generator cycle
C366	1	1	SYNC response

3.2 9400

parameter	value set during startup		
	SYNC generator: PLC or SYNC-device	SYNC generator: 1st drive	
16#1006, 16#60C2, C1121	cycle time cycle time		SYNC cycle
16#1005	16#80	16#4000 0080	Address mode
16#1400-16#1AFF			PDO mapping
16#6092	16#10000		feedconst_e4

4 CAN-Traffic

The following values apply for ECSxM and 9400 Stateline under the precondition that only one pair of set and actual values (e.g. SetPosition, ActPosition) is transmitted. If additional cyclic data is used, the Can traffic will increase and less drives can be controlled with the same cycle time.

base load:

Telegram	*Data bytes*	*Bit length*	*125 kBit/s*	*250 kBit/s*	*500 kBit/s*	*1 MBit/s*
SYNC	0	47	0,376 ms	0,188 ms	0,094 ms	0,047 ms
SDO	8	111	0,888 ms	0,444 ms	0,222 ms	0,111 ms
Overall			1,264 ms	0,632 ms	0,316 ms	0,158ms

per drive :

Telegram	*Data bytes*	*Bit length*	*125 kBit/s*	*250 kBit/s*	*500 kBit/s*	*1 MBit/s*
Control Word, set position	8	111	0,888 ms	0,444 ms	0,222 ms	0,111 ms
Status Word, actual position	8	111	0,888 ms	0,444 ms	0,222 ms	0,111 ms
overall			1,776 ms	0,888 ms	0,444 ms	0,222 ms

According to that, the following table shows the maximum number of drives per cycle time:

max. number of drives	125 kBit/s	250 kBit/s	500 kBit/s	1 MBit/s
1 ms	0	0	1	3
2 ms	0	1	3	8
3 ms	0	2	5	12
4 ms	1	3	8	16
5 ms	2	5	10	20
6 ms	2	6	12	24
8 ms	3	8	16	32